The SkyLLH framework for IceCube point-source search

Tomas Kontrimas[1*] and Martin Wolf[1] on behalf of the IceCube Collaboration
[1]Technical University of Munich

Introduction

- Hypothesis tests based on unbinned log-likelihood (LLH) functions are a common technique used in multi-messenger astronomy, including IceCube's neutrino point-source searches
- The open-source Python3-based tool "**SkyLLH**" provides a modular framework for implementing custom likelihood functions and executing log-likelihood ratio hypothesis tests
- $$\mathcal{L}\left(n_s, \vec{p}_s | D\right) = \prod_{i=1}^{N} \left[\frac{n_s}{N} S_i\left(\vec{p}_s\right) + \left(1 - \frac{n_s}{N}\right) B_i\right]$$

Structure

- The SkyLLH framework is split into two packages:
  - github.com/icecube/skyllh
    contains open-source code
  - github.com/icecube/i3skyllh
    contains private IceCube specific code

skyllh:

- core ← classes defining the detector independent LLH framework
- i3 ← specific classes (derived from core) for IceCube detector
- physics ← definitions of source hypothesis
- plotting ← utility classes for plotting PDFs and PDF ratios

i3skyllh:

- analyses ← collection of pre-defined SkyLLH IceCube analyses
- datasets ← collection of pre-defined IceCube datasets

- The core class structure is tied to the mathematical objects of the likelihood (ratio) function

- Well defined interfaces between classes allow an easy interchange or expansion of components, such as:
  - Minimizer implementation
  - Signal/background generation method
  - Source hypothesis
  - Event selection method

Kernel Density Estimator (KDE)

- Support for KDE PDFs is essential for a new search for neutrino point-sources with IceCube analysis likelihood construction [1]
- The new likelihood construction requires generation of KDE based signal PDFs on a power-law flux index (γ) grid
- The SkyLLH framework was extended to provide linear and parabola grid manifold interpolation on the γ grid
- Cache interpolated values to speed up likelihood minimization when the spectral index value stays similar

Top level analysis example

- Specific analysis definition is "hidden" in `create_analysis` function

```python
import numpy as np

from skyllh.core.random import RandomStateService
from skyllh.physics.source import PointLikeSource
from i3skyllh.datasets import data_samples
from i3skyllh.analyses.kdepdf_mcbg_ps.analysis import create_analysis

dsc = data_samples['NorthernTracks_v005p00_KDE_PDF_v007'].create_dataset_collection()
datasets = dsc.get_datasets('IC86_2011-IC86_2019')
source = PointLikeSource((np.radians(77.358), np.radians(5.693)))

analysis = create_analysis(datasets, source)

rss = RandomStateService(0)
(TS, fitparam_dict, status) = analysis.unblind(rss)
```

Summary

- The SkyLLH framework is being developed within the IceCube collaboration as a standard tool to search for neutrino emitting sources in the Universe
- Code structure allows plug-n-play concepts for building likelihood analysis
- The implementation of generalized concepts in terms of source hypothesis and hypothesis parameter definition makes the framework easy to use also for searches of other messenger particles in other experiments

References:
[1] **IceCube** Collaboration *PoS* **ICRC2021** (2021) 1138